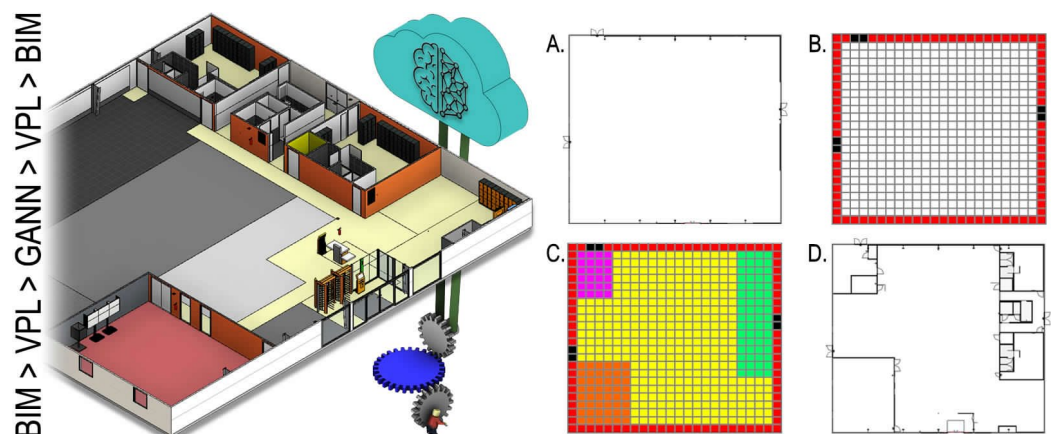
Un approccio euristico alla progettazione. Transizioni da algoritmi generativi a modelli parametrici

Massimiliano Lo Turco
Andrea Tomalini
Jacopo Bono

Abstract

Il contributo affronta criticamente il tema della generazione automatica di layout distributivi rispettosi di una coerenza topologico/spaziale e ottenuti in tempi estremamente rapidi. Questa attività è di assoluto interesse all'interno di diverse comunità scientifiche che collaborano proficuamente tra loro: dagli esperti IT, agli studiosi delle aree della Rappresentazione e della Composizione Architettonica e Urbana per citare i principali. Per esplorare alcuni approcci possibili, inerenti alla generazione di *layout* basati su immagini, su grafi, su *layout* specializzati e relative ibridazioni ci si avvale delle potenzialità euristiche del disegno digitale. Particolare attenzione è dedicata agli approcci generativi mediante l'allenamento di reti GANN (Generative Adversarial Neural Network). Il contributo illustra i primi esiti del progetto di ricerca che vede la collaborazione tra Politecnico di Torino, Links Foundation e la società olandese di palestre Basic-Fit. In una prima fase il progetto analizza criticamente lo stato dell'arte degli algoritmi di Intelligenza Artificiale impiegati in architettura; successivamente, per dimostrare le potenzialità di tali strumenti è stato progettato un primo prototipo che rispetta i principali vincoli e criteri progettuali derivanti dall'esperienza della Società ed è inoltre caratterizzato da un elevato livello di interoperabilità nella generazione delle soluzioni selezionate in ambiente BIM.

Parole chiave
intelligenza artificiale, visual programming language, design generator, building information modeling, grafi

I passaggi salienti del *workflow*: dal modello BIM al *file* vettoriale 2D; dal *file* vettoriale al *file raster*; definizione dei vincoli progettuali; dal *file raster* all'elaborato vettoriale; dal file vettoriale 2D al modello BIM. Elaborazione grafica di A. Tomalini.

doi.org/10.3280/oa-1016-c444

Introduzione

Tra i temi più indagati degli ultimi anni nell'ambito della composizione architettonica e urbana si distinguono i diversi approcci che tentano di utilizzare tecniche di Intelligenza Artificiale per la rapida generazione di innumerevoli scenari progettuali. Gli ambiti di applicazione sono molteplici e variano dal macro al micro, ossia dall'utilizzo in ambito urbano e territoriale alla definizione di proposte distributive di singole unità abitative [Oxman 2017]. Nel contributo si ragiona sull'illustrazione di approcci innovativi per la generazione di planimetrie a scala architettonica, nel rispetto di vincoli geometrici e di caratteri distributivi definiti dall'utente. Molte ricerche di seguito brevemente descritte hanno lo scopo di chiarire limiti e potenzialità di approcci attualmente disponibili, prefigurando le potenzialità e la scalabilità dei prodotti proposti. Il saggio si articola in una prima parte che delinea lo stato dell'arte, tracciando sinteticamente i contorni di uno scenario complesso, esito di approcci differenti, denunciando al contempo limiti e opportunità [Weber et al. 2022]. Nel paragrafo successivo si introduce la realtà lavorativa della società con cui si è attivato il progetto di ricerca, riportando sinteticamente il quadro esigenziale e le motivazioni che hanno guidato il gruppo di lavoro nella definizione del prototipo, descritto nel paragrafo successivo. Il contributo si conclude con la descrizione dei successivi sviluppi su cui il gruppo di ricerca sta attualmente lavorando attraverso l'attivazione di una seconda collaborazione.

Stato dell'arte: differenti approcci, tra passato e presente

Il contributo si concentra sulle tecniche e sui metodi di 'generazione automatica' delle forme. Prima di introdurre il tema è opportuno un breve richiamo al passato, citando gli esperimenti del Vignola [Vignola 1562] per giungere al paradigma illuminista formulato da J.N.L. Durand sulla tipologia morfologica funzionale, passando per gli studi degli *Shape Grammars* della seconda metà del XX secolo, fino alle tendenze contemporanee dell'architettura parametrica (fig. 1).
Il rapporto tra regole matematiche e attributi geometrici traducibili rapidamente in proposte progettuali non è un tema nuovo; gli studi dell'architetto Luigi Moretti e del matematico Bruno de Finetti, fondatori nel 1957 dell'IRMOU (Istituto di Ricerca Matematica e Operativa per l'Urbanistica), miravano a sviluppare teorie sull'architettura parametrica basate sull'applicazione di concetti matematici alla progettazione, seguendo un approccio empirico alle problematiche delle forme [Bucci, Mulazzani 2000].
I metodi di creazione automatica di *layout* spaziali furono introdotti negli anni '70 come speculazioni artistiche sul futuro ruolo dei computer e dell'Intelligenza Artificiale in architettura. Yona Friedman propose il 'Flatwriter' [Friedman 1971] per generare soluzioni distributive di appartamenti in grado di soddisfare le preferenze d'uso di tutti i soggetti coinvolti nei progetti di edilizia cooperativa. L'automazione del processo di creazione delle planimetrie era vista come uno strumento per la progettazione partecipata. Cedric Price propose il 'Generator', ossia un'unità spaziale riconfigurabile basata su *voxel* riconfigurabile dagli utenti, elaborando diverse soluzioni. Entrambe le proposte pre-computazionali lavoravano su planimetrie progettate da un kit modulare di parti in un processo di tipo *bottom-up*.
Alla fine degli anni Settanta, George Stiny iniziò a lavorare all'analisi e alla riproduzione dei *layout* degli edifici attraverso le cosiddette grammatiche parametriche delle forme [Stiny, Eizenberg 1980]. Comprendendo i principi compositivi e le qualità spaziali di edifici esistenti, come le ville palladiane [Stiny, Mitchell 1978] o le case nella prateria di Frank Lloyd Wright [Koning 1981], i modelli classificati potevano essere utilizzati per ricreare edifici dello stesso tipo. Analogamente, Christopher Alexander rappresentò le relazioni spaziali nelle planimetrie architettoniche tradizionali attraverso grafi relazionali e strutture ad albero [Alexander 1965]. Queste astrazioni costituirono i primi passi necessari per automatizzare la generazione di planimetrie e continuano a costituire riferimenti fondamentali negli approcci computazionali contemporanei. Questo tipo di ricerche portò allo sviluppo di strumentazioni in cui le opzioni di proget-

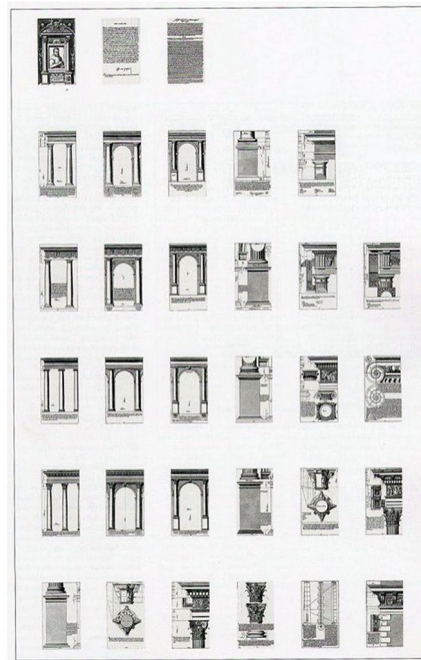
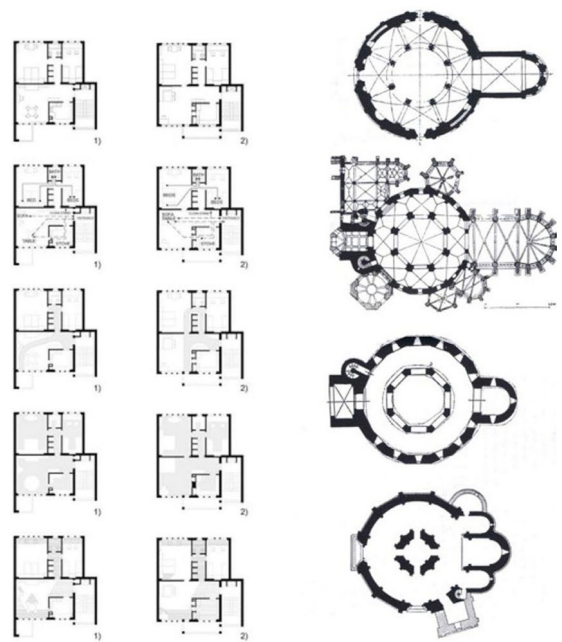
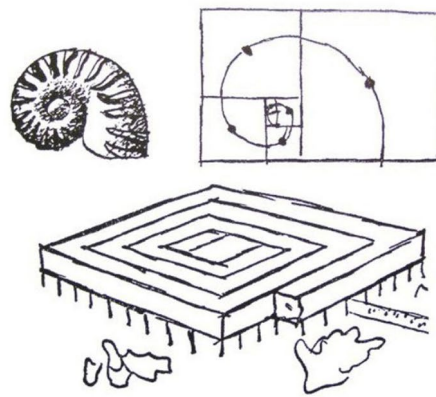
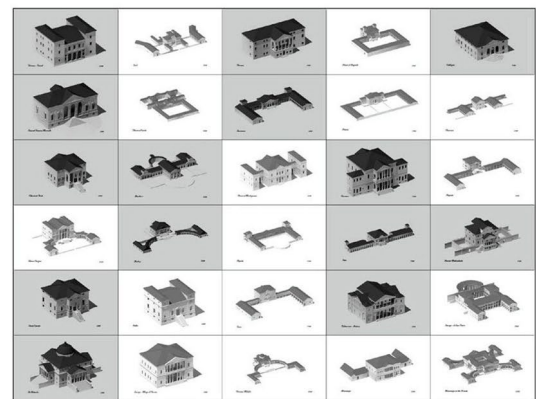
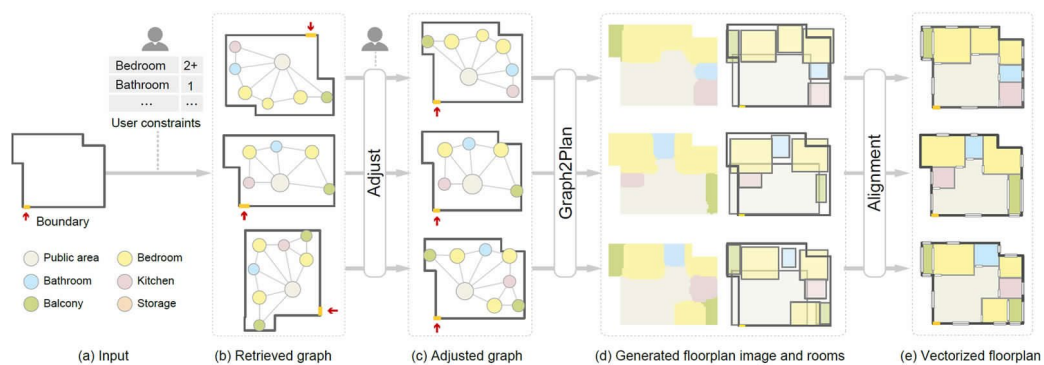



Fig. 1. Riferimenti culturali: (a) l'Esperimento del Vignola; (b) studi sulla tipologia morfologica funzionale introdotti da J.N.L. Durand; (c) dalla morfogenesi naturale a quella architettonica; (d) esempi dagli *Shape Grammars* (d). Elaborazione M. Lo Turco.

tazione generate automaticamente possono integrare i processi di progettazione manuale convenzionali, offrendo non una singola proposta ma una famiglia di soluzioni utili per successivi approfondimenti progettuali [Heumann, Davis 2020; Weber, Mueller, Reinhart 2022].

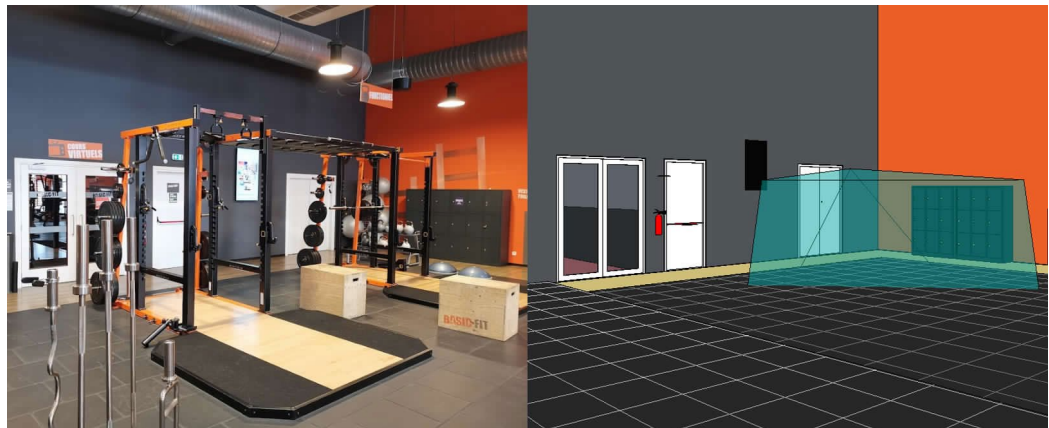
Gli approcci maggiormente in uso

In letteratura si distinguono architetture generative basate sulle immagini, sui grafi e infine i modelli specializzati. Un approccio semplice per generare un *layout* è quello di rappresentarlo come un'immagine, utilizzando architetture generative tradizionali per la sintesi di immagini [Para et al. 2021]. L'approccio più promettente è quello delle reti GANN (*Generative Adversarial Neural Network*) [Goodfellow et al. 2020; Zhang et al. 2019]. Il limite di questo approccio riguarda l'estrema difficoltà nell'identificare e, conseguentemente, restituire forme specifiche che devono preservare determinate proprietà geometriche (archi di circonferenza, ellissoidi, poligoni con specifiche proprietà, definizione di angoli specifici, ecc.).
La bibliografia dimostra come approcci puramente basati sui grafi generano principalmente la topologia del grafo, ma sono più deficitari sulla conversione dell'esito in attributi spaziali [Wang et al. 2018; You et al. 2018].

Fig. 2. Graph2Plan può generare *layout* progettuali partendo da un perimetro e da strutture a grafo (a-b). Il sistema consente di modificare il numero delle stanze (c) e le loro connessioni (d). Come output si ottengono differenti planimetrie. Hu et al. 2020, fig. 1, p. 1.

In generale tale approccio è problematico in quanto la generazione di un *layout* che ha come input un perimetro sempre differente presenta vincoli ferrei da aggiornare caso per caso. Sono inoltre presenti molteplici vincoli tra gli stessi elementi che devono comporre la pianta: l'adiacenza di determinati spazi ad alcune zone; rapporti di vicinanza e lontananza tra aree differenti; l'utilizzo di determinati componenti al variare delle condizioni a contorno degli stessi, per citare i principali. I metodi più recenti risolvono parzialmente questi problemi richiedendo una guida manuale come input o una post-elaborazione manuale. Recentemente sono stati pubblicati tre lavori che affrontano queste tematiche [Wu et al. 2019; Hu et al. 2020; Nauata et al. 2020]. Sebbene questi lavori producano spesso planimetrie a prima vista ben strutturate, presentano alcuni vincoli: tutti lavorano solo su ambienti con muri perpendicolari gli uni agli altri e tutti richiedono controlli ed eventuali correzioni in fase di post-produzione; RPLAN e Graph2Plan richiedono che sia dato il contorno della planimetria; RPLAN posiziona le porte utilizzando un approccio manuale che non viene recepito dall'algoritmo; HouseGAN non genera la connessione tra le stanze che sarebbe data dalle porte; HouseGAN e Graph2Plan richiedono come input il numero di stanze, i tipi di stanze e la loro topologia sotto forma di un grafo di adiacenza (fig. 2).
Ulteriori ricerche propongono modelli che scompongono il problema in più step producendo soluzioni più complesse. In particolare, il modello creato dal gruppo guidato da Waming Para, identifica gli ambienti e gli elementi da collocare come nodi di un grafo; in un secondo momento si inseriscono nel sistema i vincoli a cui si applica un processo di ottimizzazione per identificare la soluzione ideale [Para et al. 2021].

Non solo residenziale: la collaborazione con la società Basic-Fit

Basic-Fit è una delle più grandi società di palestre a livello europeo. Grazie a prezzi molto competitivi e a un alto livello di automazione con sorveglianza operante solo da remoto, la società si sta affermando come leader nel settore. È attualmente presente in sei nazioni con oltre mille club aperti. La società è in continua crescita grazie a processi di standardizzazione attuati mediante processi BIM per la progettazione, costruzione e successiva gestione dei locali (fig. 3). Entro il 2030 la società ha pianificato di triplicare il numero di palestre. Per raggiungere questo ambizioso obiettivo risulta strategico investire in attività di ricerca che consentano di ottimizzare la produttività degli uffici che progettano, appaltano e gestiscono i vari club.
Da queste premesse nasce la collaborazione con il Dipartimento di Architettura e Design del Politecnico di Torino (DAD) e la Links Foundation. L'obiettivo di questa prima collaborazione è stata un'attenta analisi dello stato dell'arte e la successiva strutturazione di un primo prototipo che dimostrasse le potenzialità di tali strumenti applicati ai *layout* dei club di Basic-Fit.

Fig. 3. Abbeville Rue Menchecourt. Club di Basic-Fit. Confronto tra reale (sinistra) e modello BIM (destra). Elaborazione di J. Bono.

Il prototipo

Il prototipo presentato si contraddistingue per una forte interoperabilità tra diversi ambienti di modellazione e programmazione. Il gruppo PoliTo si è occupato della preparazione dei dati da comunicare al modello di AI (BIM> vettoriale>*file raster*), della definizione dei vincoli progettuali da trasmettere al modello AI, e infine, della conversione degli esiti del modello AI in componenti edilizi in ambiente BIM (*file raster*>vettoriale>BIM). Il gruppo di Links Foundation si è concentrato sulla rielaborazione del modello di Graph2Plan e sulla codifica delle *Application Programming Interface* (API) tra il modello AI e l'ambiente VPL (fig. 4). Nel rispetto di tempi molto ristretti si è optato per utilizzare un modello già allenato in quanto più flessibile e di più facile integrazione con gli altri applicativi.
Si riportano di seguito i principali passaggi.
a) Da modello BIM a *file* vettoriale 2D.
il perimetro degli edifici è acquisito dalla piattaforma BIM attraverso codici sviluppati in ambiente VPL. Il codice seleziona i componenti edilizi che identificano il perimetro leggendo le fasi temporali dei componenti edilizi e parametri di progetto compilati dall'operatore in fase di modellazione. Successivamente si classificano le bucature in base alla loro funzione (ingresso, uscita di sicurezza, ingresso secondario). Questi elementi edilizi vengono convertiti in elementi bidimensionali (muri=linee; bucature=punti) (fig. 5).
b) Da vettoriale a *raster.*
Gli elementi bidimensionali sono successivamente traslati e scalati (i valori di rimappatura delle coordinate sono sempre differenti e vengono memorizzati dall'algoritmo) per occupare l'area maggiore all'interno di una griglia quadrata composta a sua volta di 256 x

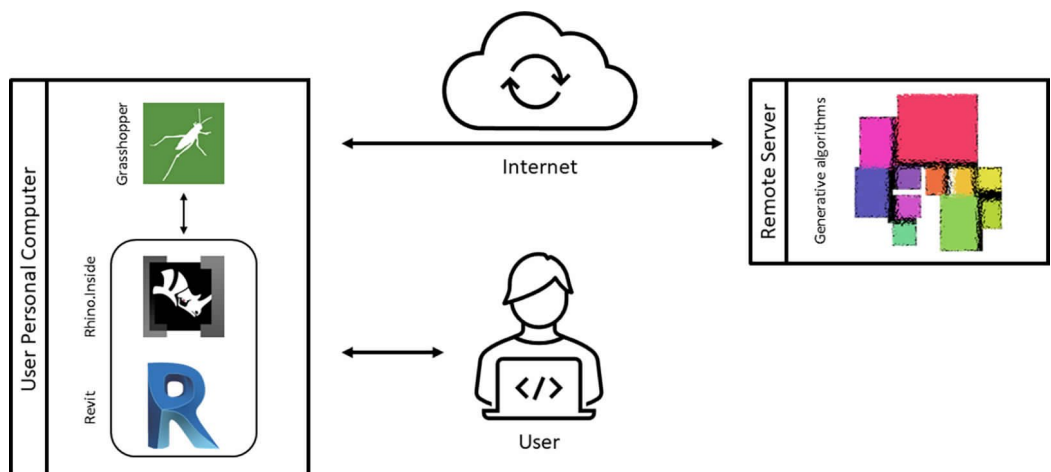



Fig. 4. Schema concettuale di funzionamento del prototipo presentato. Elaborazione di A. Tomalini.

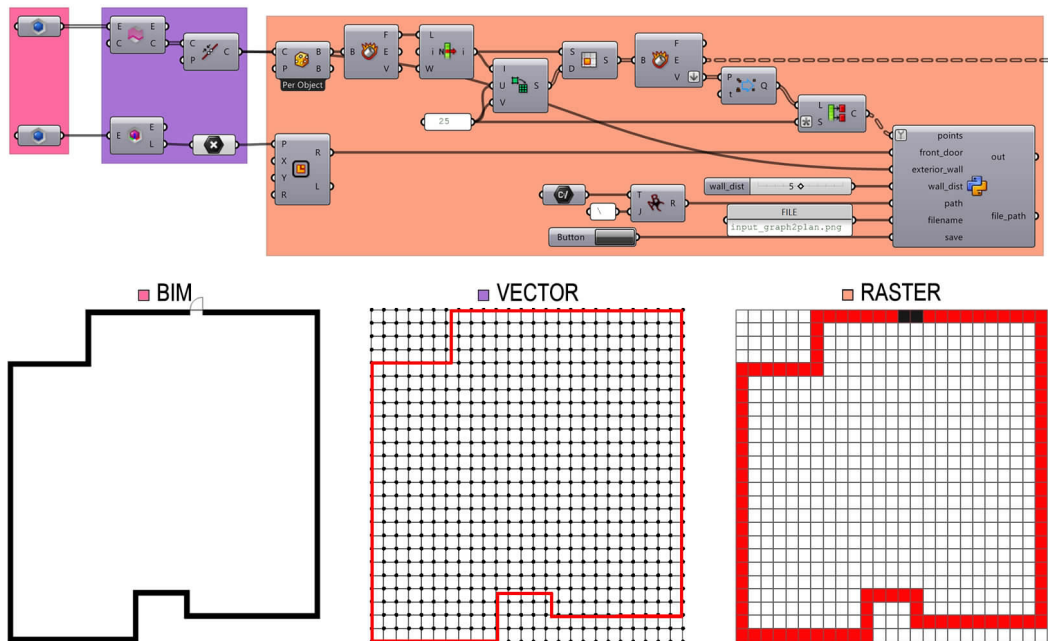



Fig. 5. Codice semplificato che caratterizza le prime operazioni del *workflow* presentato. Selezione del perimetro e dell'ingresso in ambiente BIM. Conversione in elementi vettoriali. Creazione dell'immagine raster da trasmettere al *framework custom* di Graph2Plan. Elaborazione di A. Tomalini.

256 quadrati. La posizione delle linee e dei punti rispetto alla griglia definisce quali *pixel* sono colorati e quali no e le informazioni associate a ciascuno di essi determinano il colore del *pixel*. La matrice risultante viene poi convertita in un *file raster* (*.png) (fig. 5).

c) Definizione dei vincoli progettuali.

Come anticipato, Graph2Plan è un modello che funziona attraverso la definizione di un grafo: dalla sua personalizzazione si generano soluzioni progettuali differenti. Per contrarre il più possibile i tempi, si è deciso di non allenare un nuovo modello sulla base di grafi rappresentanti i club. È stato quindi necessario adattare il sistema esistente, pensato per un uso residenziale, adattandolo alla definizione di *layout* per palestre.

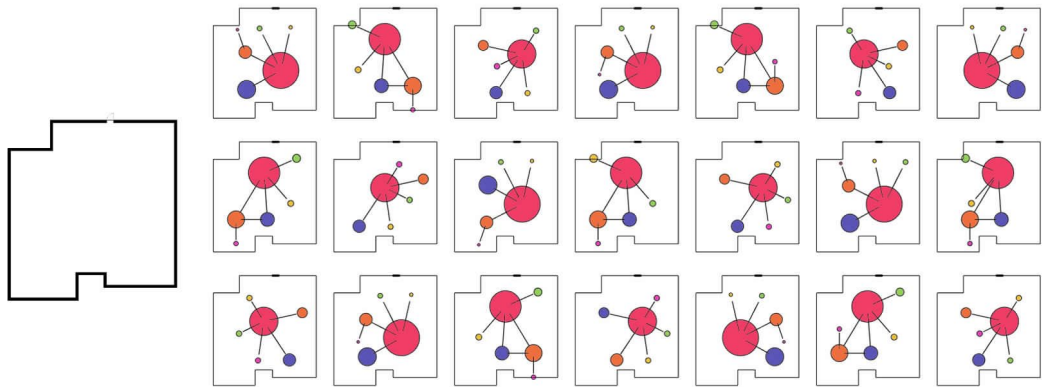
Vista l'esperienza di Basic-Fit è stato facile identificare le zone che caratterizzano i club, le loro funzioni, le metrature necessarie, i relativi rapporti di connessione, i rapporti dimensionali ecc. Sulla base di queste specifiche è stato analizzato il *dataset* dei grafi di *training* di Graph2Plan, composto da più di ottantamila *item*. Pur essendo destinato a una funzione d'uso differente è stato possibile identificare dei grafi che, con un buon livello di approssimazione, potessero comunque rappresentare le proporzioni di alcuni club. Dal *dataset* iniziale sono stati estrapolati poco meno di dodici mila grafi. Seppur costituisca una casistica limitata, questa selezione rappresenta i possibili grafi che l'utente avrebbe potuto concepire.

Il processo di definizione di un nuovo grafo è stato codificato in ambiente VPL: grazie all'attribuzione di punteggi percentuali, l'algoritmo restituisce un set di soluzioni ottimali riconducibili a un quadro esigenziale esplicitato attraverso diversi parametri, tra cui funzioni da inserire, posizione della porta d'ingresso, dimensioni delle singole zone, forma del perimetro da analizzare e così via.

Il grafo selezionato associato all'immagine *raster* rappresenta l'input con cui il modello di Graph2Plan genera diverse soluzioni progettuali. Le zone funzionali sono state rinominate e gli ambienti di dimensione troppo esigue sono esclusi dal calcolo; le piante non vengono generate in base ai metri quadri effettivi richiesti da ogni singola funzione ma bensì ai pesi percentuali che queste funzioni hanno sull'area totale (fig. 6). La velocità di calcolo del prototipo è molto ridotta poiché le operazioni sono state codificate in sequenza e non in parallelo, restituendo una differente soluzione in formato *raster* ogni venti secondi di calcolo.

d) Dal *file raster* all'elaborato vettoriale.

le soluzioni generate da Graph2Plan sono scaricate in locale e sono presentate come immagini *raster* di 256 x 256 *pixel*. Le estremità dei *pixel* colorati identificano le coordinate delle linee bidimensionali e il punto di inserimento delle aperture. Grazie ai valori di traslazione e scalatura delle geometrie bidimensionali è possibile riportare le geometrie alla forma e

Fig. 6. Fase di elaborazione in Graph2Plan. Selezione dello schema a grafo che meglio risponde alle necessità progettuali. A ognuna di queste planimetrie è associato un punteggio percentuale per facilitare la scelta dell'operatore. Elaborazione di A. Tomalini.

dimensioni originali, seppure con alcune approssimazioni, poiché il processo di rimappatura delle coordinate in un dominio di 256 × 256 non è adatto a rappresentare tutti i valori decimali di una determinata coordinata (fig. 7).

e) Dal *file* vettoriale 2D al modello BIM.

L'ultima porzione di algoritmo, sviluppato in ambiente VPL, associa a ogni elemento bidimensionale il corretto componente edilizio, generato automaticamente all'interno della piattaforma BIM attraverso un apposito algoritmo (fig. 7).

Conclusioni e prossimi sviluppi

L'applicazione dell'Intelligenza Artificiale per la strutturazione di *layout* progettuali è una tematica che accomuna molteplici discipline anche molto distanti tra loro. L'utilizzo degli algoritmi di apprendimento automatico e i progressi degli strumenti computazionali integrati nei tradizionali ambienti di modellazione geometrica utilizzati nell'ambito della progettazione architettonica possono contribuire a colmare il divario interdisciplinare tra gli architetti e gli esperti di *Information Technologies*, nell'applicazione di conoscenze specifiche derivanti dai campi di indagine dell'Intelligenza Artificiale. L'analisi dello stato dell'arte restituisce un contesto estremamente dinamico, in termini di sviluppo di nuovi strumenti proposti sia dal comparto industriale sia dagli interessanti spunti derivanti dalla ricerca accademica. Il prototipo proposto ha prodotto esiti tangibili nonostante alcune criticità che verranno

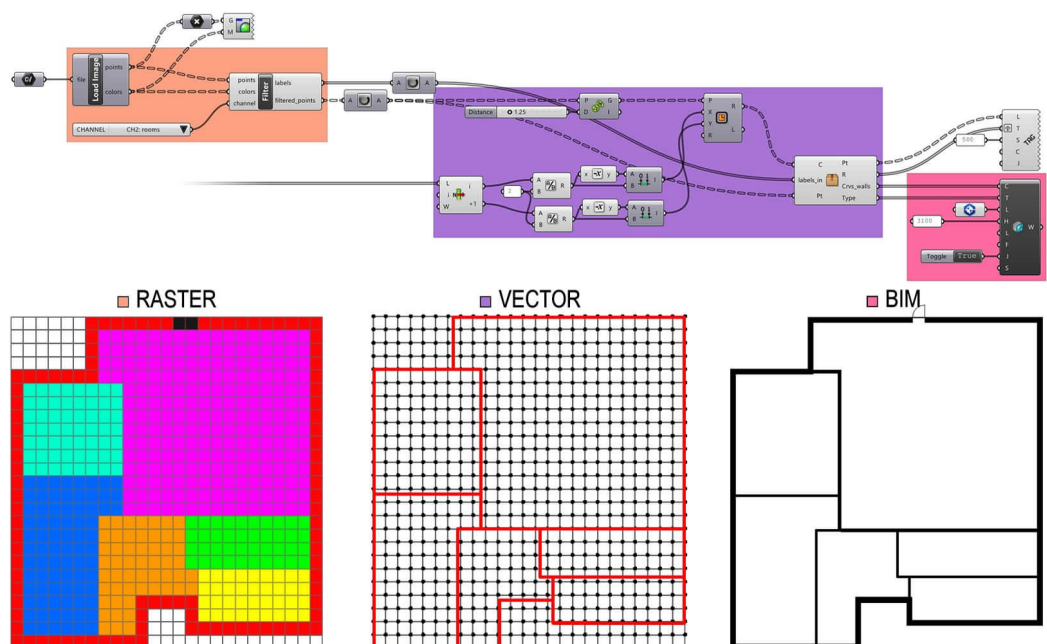



Fig. 7. Codice semplificato delle ultime operazioni del *workflow* presentato. L'immagine elaborata mediante Graph2Plan viene acquisita in ambiente VPL e, grazie ai valori RGB dei *pixel* vengono estratti i componenti edilizi e riportati in ambiente BIM – Rhino Inside Revit. Elaborazione di A. Tomalini.

risolte nel prosieguo della ricerca. Nella nuova collaborazione in essere gli obiettivi sono ancora più ambiziosi: il gruppo di ricerca infatti è attualmente impegnato nella codifica di un modello che risolva le seguenti problematiche: l'applicazione del tool su progetti pluripiano; la gestione di forme complesse; la verifica dell'interpiano in funzione delle diverse funzioni associate alle aree; l'opportunità di escludere alcune aree dal calcolo (pilastri, vani tecnici, ecc..). Se tali obiettivi verranno traguardati il prodotto finale potrà effettivamente costituire parte integrante del *workflow* di progettazione di Basic-Fit.

Crediti

Del presente contributo, di cui gli autori hanno condiviso l'impianto metodologico, Massimiliano Lo Turco ha redatto 'Introduzione' e 'Conclusioni e prossimi sviluppi'; Andrea Tomalini ha redatto 'Gli approcci maggiormente in uso' e 'Il prototipo'; Jacopo Bono ha redatto 'Stato dell'arte: differenti approcci, tra passato e presente' e 'Non solo residenziale: la collaborazione con la società Basic-Fit'.

Riferimenti bibliografici

Alexander C. (1965). A city is not a Tree, Parts 1 & 2. In *Architectural Forum*, n. 122, pp. 58-62.

Barozzi Jacopo da Vignola (1562). *La Regola delli cinque ordini dell'architettura di Iacopo Barozzi parda Vignola*. Roma.

Bucci F., Mulazzani M. (2000). *Luigi Moretti: Opere e scritti*. Milano: Electa.

Friedman Y. (1971). Flatwriter: choice by computer. In *Progressive Architecture*, n. 3, pp. 89-101.

Goodfellow I., Pouget-Abadie, J., Mirza M., Xu B., Warde-Farley D., Ozair S., Bengio Y. (2020). Generative adversarial networks. In *Communications of the ACM*, vol. 63, n. 11, pp. 139-144.

Heumann A., Davis D. (2020). Humanizing Architectural Automation: A Case Study in Office Layouts. In C. O. Gengnagel, J. Baverel, M. Burry, T. Ramsgaard, S. Weinzierl (a cura di). *7th Design Modelling Symposium "Impact: Design with All Senses". Conference proceedings*. Berlino, settembre 2019, pp. 662-670. Cham: Springer.

Hu R., Huang Z., Tang Y., van Kaick O., Zhang H., Huang H. (2020). Graph2plan: Learning floorplan generation from layout graphs. In *ACM Transactions on Graphics (TOG)*, vol. 39, n. 4, pp. 118:1-118:14.

Koning H., Eizenberg J. (1981). The language of the prairie: Frank Lloyd Wright's prairie houses. In *Environment and Planning. B: Planning & Design*, vol. 8, n. 3, pp. 295-323.

Nauata N., Chang K. H., Cheng C. Y., Mori G., Furukawa Y. (2020). House-gan: Relational generative adversarial networks for graph-constrained house layout generation. In *Computer Vision-ECCV 2020: 16th European Conference. Conference proceedings*. Glasgow, UK, 23-28 agosto 2020. Part I 16, pp. 162-177.

Oxman R. (2017). Thinking difference: Theories and models of parametric design thinking. In *Design studies*, n. 52, pp. 4-39.

Para W., Guerrero P., Kell,y T., Guibas L., Wonka, P. (2021). Generative Layout Modeling using Constraint Graphs. *IEEE/CVF International Conference on Computer Vision. Conference proceedings*. Montreal, Canada, 11-17 ottobre 2021, pp. 6690-6700. Computer Vision Foundation.

Stiny G. (1980). Introduction to shape and shape grammars. In *Environment and Planning. B: Planning & Design*, vol. 7, n. 3, pp. 343-351.

Stiny G., Mitchell W. J. (1978). The Palladian Grammar. In *Environment and Planning B: Planning and Design*, vol. 5, n. 1, pp. 5-18.

Wang H., Wang J., Zhao M., Zhang W., Zhang F., Guo M. (2018). Graphgan: Graph representation learning with generative adversarial nets. In *AAAI conference on artificial intelligence. Conference proceedings*. New Orleans, Louisiana, USA, 2-7 febbraio 2018, vol. 32, n. 1, pp. 2508-2515.

Weber R. E., Mueller C., Reinhart C. (2022). Automated floorplan generation in architectural design: A review of methods and applications. In *Automation in Construction*, vol. 140, pp. 104-385.

Wu W., Fu X. M., Tang R., Wang Y., Qi Y. H., Liu L. (2019). Data-driven interior plan generation for residential buildings. In *ACM Transactions on Graphics (TOG)*, vol. 38, n. 6, pp. 1-12.

You J., Ying R., Ren X., Hamilton W., Leskovec J. (2018). Graphrnn: Generating realistic graphs with deep auto-regressive models. In *35th International Conference on Machine Learning. Conference proceedings*. Stoccolma, Svezia, 10-15 luglio 2018, vol. 80, pp. 5708-5717.

Zhang H., Goodfellow I., Metaxas D., Odena A. (2019). Self-attention generative adversarial networks. In *36th International conference on machine learning. Conference proceedings*. Long Beach USA, 10-15 giugno 2019, pp. 7354-7363.

Autori
Massimiliano Lo Turco, Politecnico di Torino, massimiliano.loturco@polito.it
Andrea Tomalini, Politecnico di Torino, andrea.tomalini@polito.it
Jacopo Bono, Politecnico di Torino, jacopo.bono@polito.it

Per citare questo capitolo: Lo Turco Massimiliano, Tomalini Andrea, Bono Jacopo (2023). Un approccio euristico alla progettazione. Transizioni da algoritmi generativi a modelli parametrici/A Heuristic Approach to Design. Transitions from Generative Algorithms to Parametric Models. In Cannella M., Garozzo A., Morena S. (a cura di). *Transizioni. Atti del 44° Convegno Internazionale dei Docenti delle Discipline della Rappresentazione/ Transitions. Proceedings of the 44th International Conference of Representation Disciplines Teachers*. Milano: FrancoAngeli, pp. 2914-2930.

 Isbn 9788835155119

A Heuristic Approach to Design. Transitions from Generative Algorithms to Parametric Models

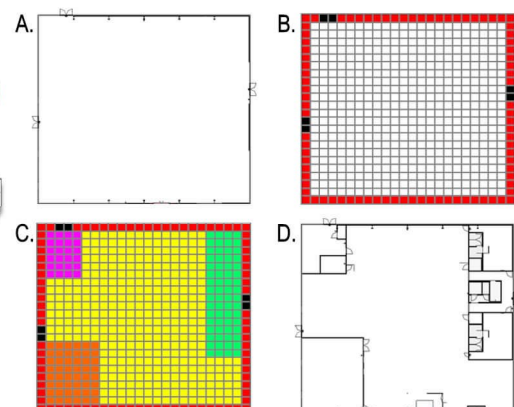
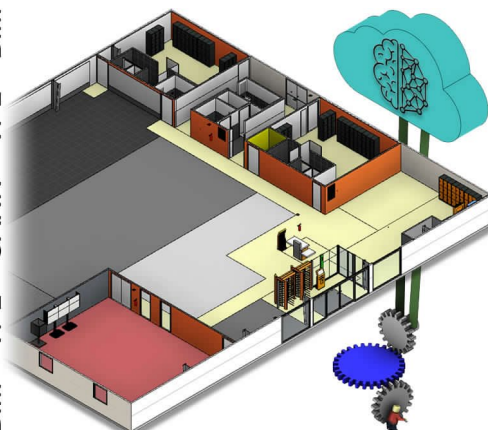
Massimiliano Lo Turco
Andrea Tomalini
Jacopo Bono

Abstract

The paper critically addresses the issue of the automatic generation of distributional layouts that respect topological/spatial coherence and are obtained exceptionally quickly. This activity is of absolute interest within several scientific communities that collaborate fruitfully with each other: from IT experts, to scholars in the areas of Architectural and Urban Representation and Composition to name the main ones. We use the heuristic potential of digital drawing to explore some possible approaches inherent in the generation of layouts based on images, graphs, specialized structures and related hybridizations. Particular attention is paid to generative processes by training Generative Adversarial Neural Network (GANN) networks. The paper outlines the first outcomes of a research project involving a collaboration between Politecnico di Torino, Links Foundation and the Dutch gymnasium company Basic-Fit. In the first phase, the project critically analyzes the state of the art of Artificial Intelligence algorithms used in architecture; then, to demonstrate the potential of these tools, a first prototype was designed that respects the primary design constraints and criteria derived from the Company's experience and is also characterized by a high level of interoperability in the generation of the selected solutions in the BIM environment.

Keywords
Artificial Intelligence, Visual Programming Language, Design Generator, Building Information Modeling, Graphs

The salient steps of the workflow: from BIM model to 2D vector file; from vector to raster; definition of design constraints; from raster file to vector work; from 2D vector file to BIM model. Graphic elaboration by A. Tomalini.

doi.org/10.3280/oa-1016-c444

Introduction

Among the most investigated topics in recent years in architectural and urban composition are the various approaches that attempt to use Artificial Intelligence techniques to generate countless design scenarios rapidly. The application areas are many and vary from the macro to the micro, from use in urban and spatial settings to the definition of distributional proposals for individual housing units [Oxman 2017]. In this contribution, we illustrate innovative approaches for generating floor plans at the architectural scale while respecting geometric constraints and user-defined distributional characters. Much of the research briefly described below aims to clarify the limitations and potential of currently available approaches, foreshadowing the potential and scalability of proposed products. The essay consists of a first part outlining the state of the art, briefly drawing the contours of a complex scenario, and the outcome of different approaches, while denouncing limitations and opportunities [Weber et al. 2022]. The following paragraph introduces the working reality of the company with which the research project was activated, briefly reporting the demanding framework and motivations that guided the working group in defining the prototype, described in the next paragraph. The contribution concludes with a description of the subsequent developments on which the research group is currently working through the activation of a second collaboration.

State of the art: different approaches, between past and present

The paper focuses on techniques and methods of 'automatic generation' of forms. Before introducing the topic, a brief reference to the past is appropriate, citing the experiments of Vignola [Vignola 1562] to arrive at the Enlightenment paradigm formulated by J.N.L. Durand on functional morphological typology, passing through the studies of Shape Grammars in the second half of the 20th century to the contemporary trends of parametric architecture (fig. 1).
The relationship between mathematical rules and geometric attributes that can be quickly translated into design proposals is not a new theme; the studies of the architect Luigi Moretti and mathematician Bruno de Finetti, founders in 1957 of the IRMOU (Institute of Mathematical and Operational Research for Urban Planning), aimed to develop theories of parametric architecture based on the application of mathematical concepts to design, following an empirical approach to the problems of forms [Bucci, Mulazzani 2000].
Methods of automatic creation of spatial layouts were introduced in the 1970s as artistic speculations on the future role of computers and Artificial Intelligence in architecture. Yona Friedman proposed the 'Flatwriter' [Friedman 1971] to generate apartment distribution solutions that meet the user preferences of all stakeholders in cooperative housing projects. Automation of the floor plan creation process was seen as a tool for participatory design. Cedric Price proposed the 'Generator' a voxel-based reconfigurable spatial unit that users can reconfigure by working out different solutions. Both pre-computational proposals worked on floor plans designed from a modular parts kit in a bottom-up process.
In the late 1970s, George Stiny began working on analyzing and reproducing building layouts through so-called parametric grammars of forms [Stiny, Eizenberg 1980]. By understanding existing buildings' compositional principles and spatial qualities, such as Palladian villas [Stiny, Mitchell 1978] or Frank Lloyd Wright's prairie houses [Koning 1981], classified models could be used to recreate buildings of the same type. Similarly, Christopher Alexander represented spatial relationships in traditional architectural floor plans through relational graphs and tree structures [Alexander 1965]. These abstractions constituted the first steps necessary to automate the generation of floor plans and continue to be critical references in contemporary computational approaches.
This research led to the development of instrumentations in which automatically generated design options can complement conventional manual design processes, offering not a single proposal but a family of solutions useful for subsequent design insights [Heumann, Davis 2020; Weber, Mueller, Reinhart 2022].

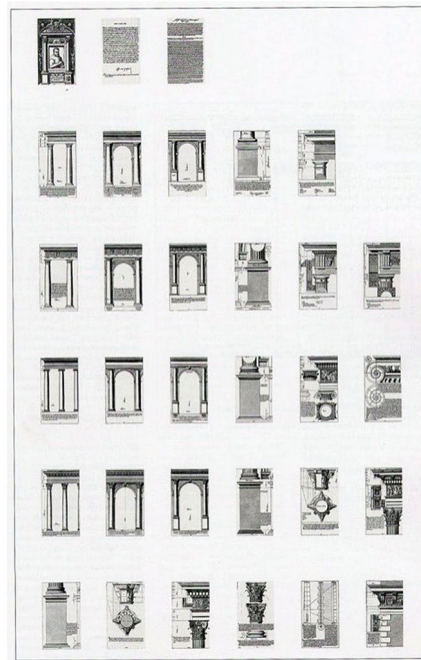
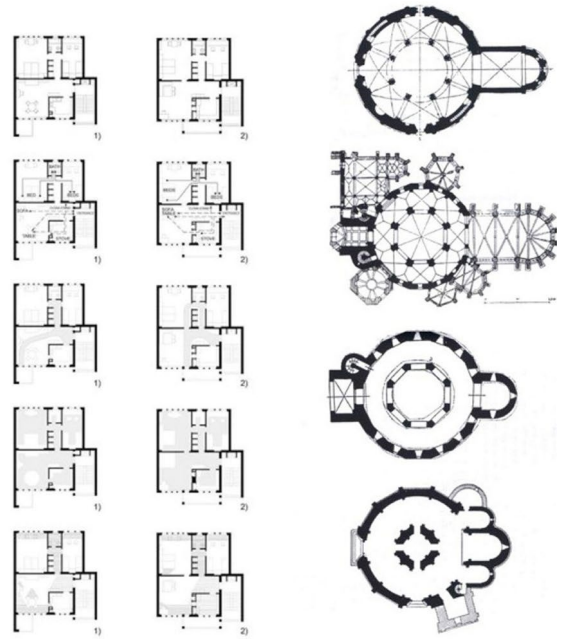
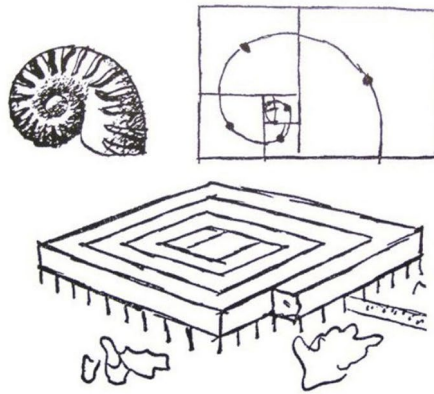
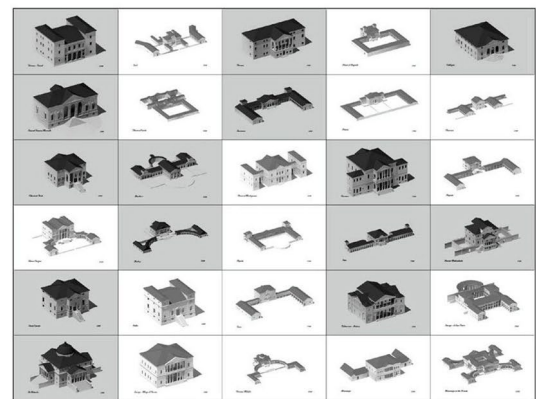
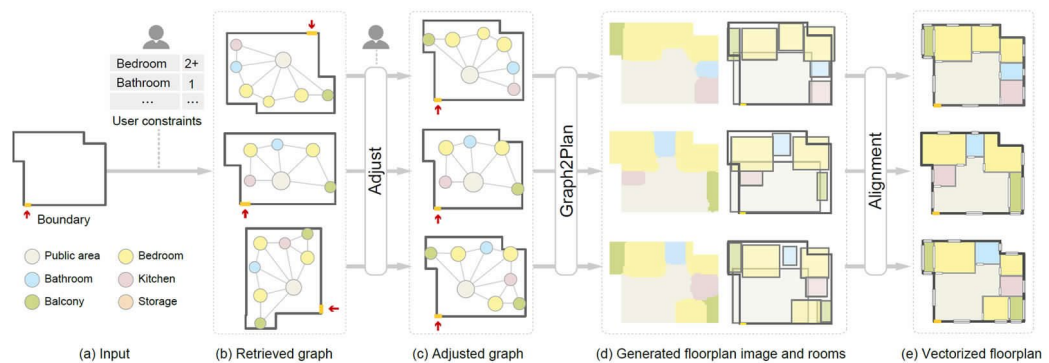



Fig. 1. Cultural references: (a) Vignola's Experiment; (b) studies on functional morphological typology introduced by J.N.L. Durand; (c) from natural to architectural morphogenesis; (d) examples taken from *Shape Grammars*. Elaboration by M. Lo Turco.

The most commonly used approaches

Generative architectures based on images, graphs and finally specialized models are distinguished in the literature. A simple approach to generating a layout is representing it as an image, using traditional generative architectures for image synthesis [Para et al. 2021]. The most promising approach is Generative Adversarial Neural Network (GANN) networks [Goodfellow et al. 2020; Zhang et al. 2019]. The limitation of this approach concerns the extreme difficulty in identifying and, consequently, returning specific shapes that must preserve certain geometric properties (circumference arcs, ellipsoids, polygons with particular properties, the definition of precise angles, etc...).
The bibliography shows how purely graph-based approaches mainly generate graph topology, but are more deficient in converting the outcome into spatial attributes [Wang et al. 2018; You et al. 2018].
In general, this approach is problematic because generating a layout that has an ever-changing perimeter as input has ironclad constraints that need to be updated on a case-by-case basis. There are also multiple constraints between the same elements that must make up the layout: the adjacency of specific spaces to certain areas; proximity and distance relationships

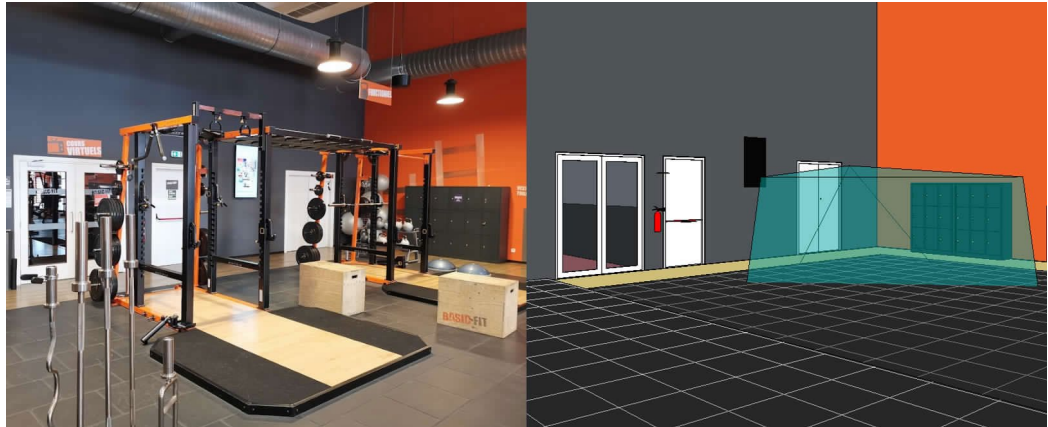
Fig. 2. Graph2Plan can generate design layouts from perimeter and graph structures (a-b). The system allows changing the number of rooms (c) and their connections (d). Different floor plans are obtained as output. Hu et al. 2020, fig. 1, p. 1.

between different areas; the use of specific components as the boundary conditions of those components changes, to name the main ones. Newer methods partially solve these problems by requiring manual guidance as input or manual post-processing. Three papers addressing these issues have recently been published [Wu et al. 2019; Hu et al. 2020; Nauata et al. 2020]. Although these jobs often produce well-structured floor plans at first glance, they have some constraints: all of them work only on rooms with walls perpendicular to each other and all require checks and possible corrections in post-production; RPLAN and Graph2Plan require that the outline of the floor plan be given; RPLAN places doors using a manual approach that is not picked up by the algorithm; HouseGAN does not generate the connection between rooms that the doors would give; HouseGAN and Graph2Plan require as input the number of rooms, the types of spaces and their topology in the form of an adjacency graph (fig. 2).
Further research proposes models that decompose the problem into multiple steps producing more complex solutions. Specifically, the model created by the group led by Waming Para, identifies the rooms and elements to be placed as nodes in a graph; in a second step, constraints are entered into the system to which an optimization process is applied to identify the ideal solution [Para et al. 2021].

Not only residential: collaboration with the Basic-Fit company

Basic-Fit is one of the largest gymnasium companies in Europe. The company is establishing itself as a leader in the industry due to very competitive prices and a high level of automation with surveillance operating only remotely. It is currently present in six countries with over a thousand clubs open. The company is continually growing through standardization processes implemented through BIM processes for the design, construction, and subsequent management of clubs (fig. 3). By 2030 the company has planned to triple the number of gyms. To achieve this ambitious goal, it is strategic to invest in research activities that will optimize the productivity of the offices that design, contract and manage the various clubs. The collaboration with the Department of Architecture and Design of the Polytechnic University of Turin (DAD) and the Links Foundation was born from these premises. The goal of this first collaboration was a careful analysis of the state of art and the subsequent structuring of an initial prototype that would demonstrate the potential of such tools applied to Basic-Fit club layouts.

The prototype

The prototype is characterized by reliable interoperability between modeling and programming environments. The PoliTo group was responsible for the preparation of the data to be communicated to the AI model (BIM>vector>raster file), the definition of the design constraints to be transmitted to the AI model, and finally, the conversion of the AI model outcomes into building components in the BIM environment (raster>vector>BIM file). The

Fig. 3. Abbeville Rue Menchecourt. The Basic-Fit Club. Comparison of actual (left) and BIM model (right). Elaboration by J. Bono.

Links Foundation team focused on reworking the Graph2Plan model and coding the Application Programming Interfaces (APIs) between the AI model and the VPL environment (fig. 4). Within very tight time constraints, they opted to use an already trained model as it was more flexible and easier to integrate with other applications.
The main steps are given below.
a) BIM model to 2D vector file.
The building perimeter is acquired from the BIM platform through codes developed in the VPL environment. The code selects the building components that identify the perimeter by reading the time steps of the building components and design parameters compiled by the operator during modeling. Next, the holes are classified according to their function (entrance, emergency exit, secondary entrance). These building elements are converted to two-dimensional elements (walls=lines; holes=dots) (fig. 5).
b) Vector to raster.
The two-dimensional elements are subsequently translated and scaled (coordinate remapping values are always different and are stored by the algorithm) to occupy the most significant area within a square grid composed of 256x256 squares. The position of the lines and points relative to the grid defines which pixels are colored and which are not, and the information associated with each determines the pixel's color. The resulting matrix is then converted into a raster file (*.png) (fig. 5).
c) Definition of design constraints.
As anticipated, Graph2Plan is a model that works through the definition of a graph: different design solutions are generated from its customization. It was decided not to train a new model based on graphs representing clubs to contract the time as much as possible. Therefore, it was necessary to adapt the existing system, designed for residential use, by adjusting it to the definition of layouts for gyms.

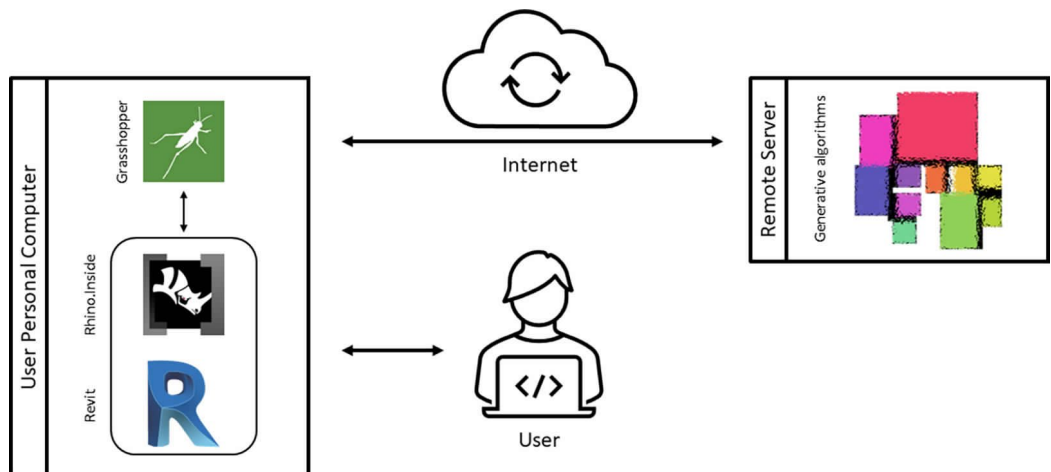



Fig. 4. Conceptual diagram of the operation of the presented prototype. Elaboration by A. Tomalini.

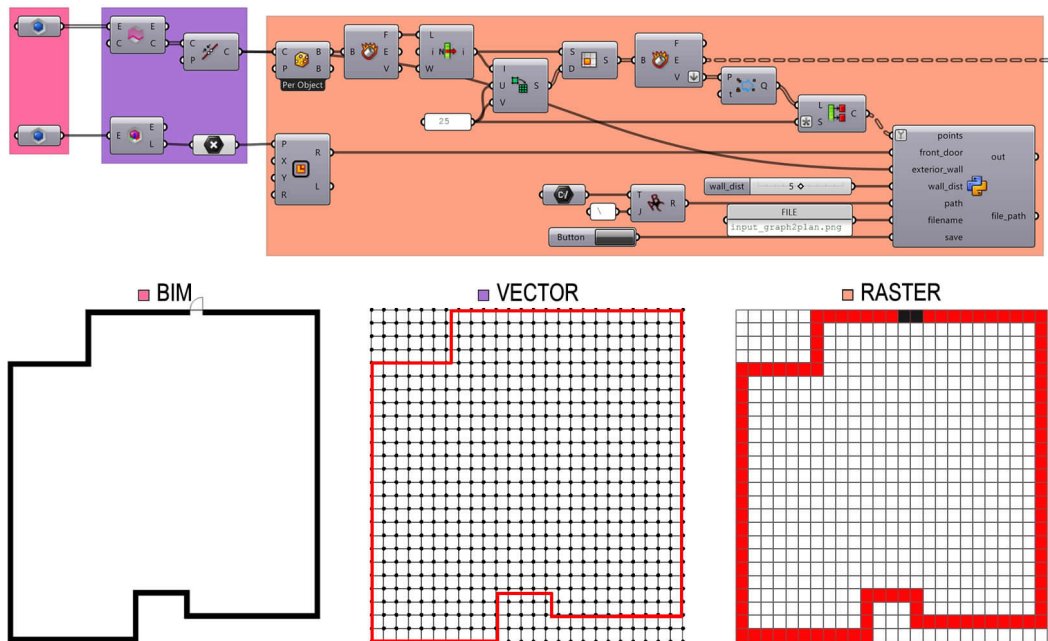
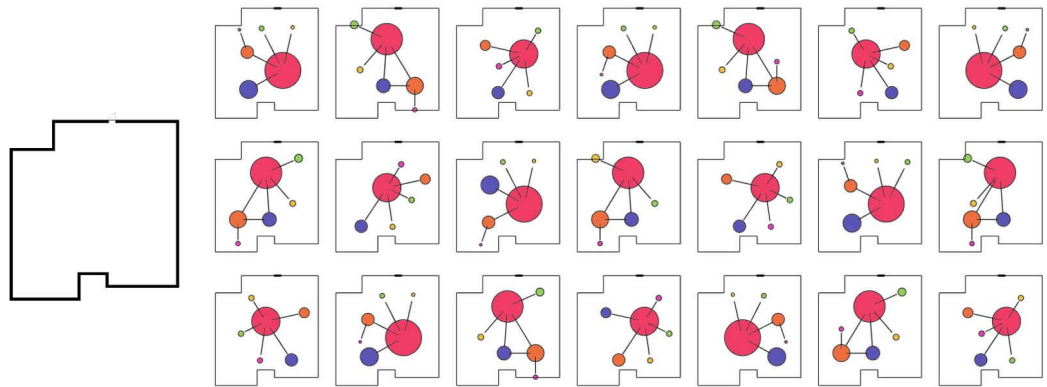



Fig. 5. Simplified code characterizing the first operations of the presented workflow. Perimeter and input selection in the BIM environment. Conversion to vector elements. Creation of the raster image to be transmitted to the Graph2Plan custom framework. Elaboration by A. Tomalini.

Given Basic-Fit's experience, it was easy to identify the zones that characterize clubs, their functions, required square footage, relative connection ratios, dimensional ratios, etc. The Graph2Plan training graph dataset, consisting of more than eighty thousand items, was analyzed based on these specifications. Although intended for a different use function, it was possible to identify graphs that, with a good level of approximation, could still represent the proportions of some clubs. Just under twelve thousand graphs were extracted from the initial dataset. While constituting a limited case study, this selection represents the possible graphs the user could have conceived.
The process of defining a new graph was coded in the VPL environment. By assigning percentage scores, the algorithm returns a set of optimal solutions that can be traced back to a demanding framework made explicit through several parameters, including: features to be included, the position of the entrance door, size of individual zones, shape of the perimeter to be analyzed, and so on.
The selected graph associated with the raster image represents the input with which the Graph2Plan model generates various design solutions. Functional zones have been renamed and rooms that are too small in size are excluded from the calculation; plans are not developed based on the actual square meters required by each feature but instead on the percentage weights these features have on the total area (fig. 6). The calculation speed of the prototype is significantly reduced because the operations were coded sequentially and not in parallel, returning a different solution in raster format every twenty seconds of calculation;
d) From raster file to vector processing.
The solutions generated by Graph2Plan are downloaded locally and are presented as raster images of 256 x 256 pixels. The ends of the colored pixels identify the coordinates of the two-dimensional lines and the insertion point of the openings. Thanks to the translation and scaling values of the two-dimensional geometries, it is possible to restore the geometries to their original shape and dimensions, albeit with some approximations, since the coordinate remapping process in a 256 x 256 domain is not suitable for representing all decimal values of a given coordinate (fig. 7).
e) From the 2D vector file to the BIM model.
The last portion of the algorithm, developed in the VPL environment, associates each two-dimensional element with the correct building component, which is automatically generated within the BIM platform through a special algorithm (fig. 7).

Fig. 6. Processing stage in Graph2Plan. Selection of the graph scheme that best meets the design needs. A percentage score is associated with these plans to facilitate the operator's choice. Elaboration by A. Tomalini.

Conclusions and next developments

The application of Artificial Intelligence for structuring design layouts is an issue that unites multiple disciplines, even those far apart. Using machine learning algorithms and advances in computational tools integrated into traditional geometric modeling environments used in architectural design can help bridge the interdisciplinary gap between architects and Information Technologies experts in applying specific knowledge derived from the fields of investigation of Artificial Intelligence. The analysis of the state-of-the-art returns to a highly dynamic context, in terms of developing new tools proposed by the industrial sector and the exciting insights from academic research.
The proposed prototype has produced tangible outcomes despite some critical issues that will be resolved in the continuation of the research. In the new collaboration in place, the goals are even more ambitious: in fact, the research group is currently engaged in the coding of a model that solves the following problems: the application of the tool on multi-story projects; the management of complex forms; the verification of the inter-story according to the different functions associated with the areas; and the opportunity to exclude some areas from the calculation (pillars, technical compartments, etc..). If these objectives are achieved, the final product can effectively form an integral part of the Basic-Fit design workflow.

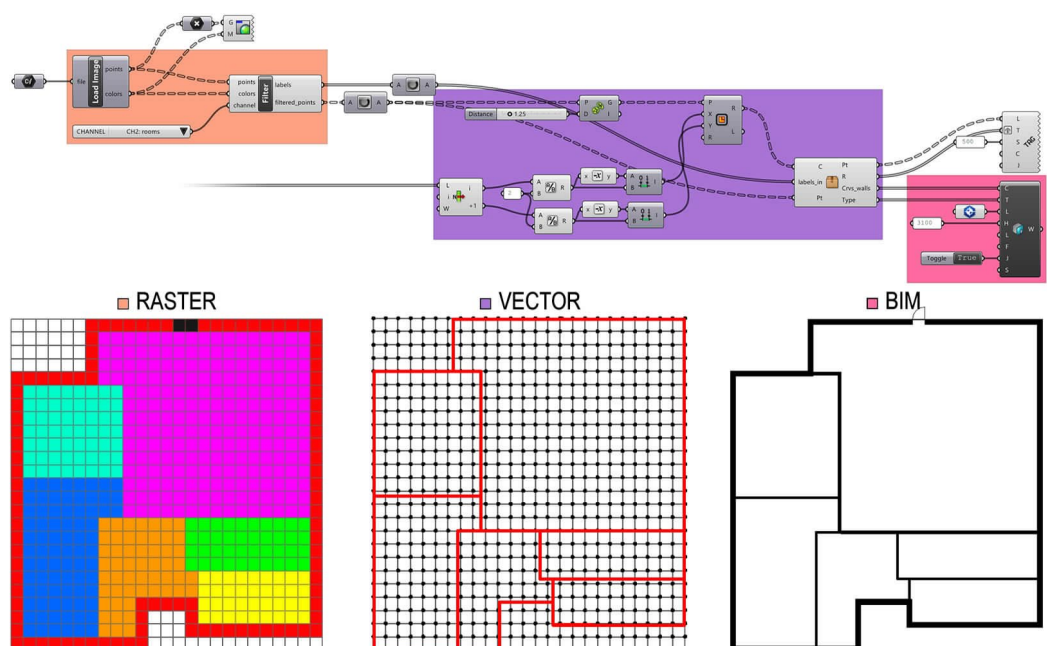



Fig. 7. Simplified code of the last operations of the presented workflow. The image processed using Graph2Plan is acquired in the VPL environment. Thanks to RGB values of the pixels, the building components are extracted and reported in the BIM environment - Rhino Inside Revit. Elaboration by A. Tomalini.

Credits

For this paper, whose authors shared the methodological framework, Massimiliano Lo Turco drafted 'Introduction' and 'Conclusions and next developments'; Andrea Tomalini drafted 'The most commonly used approaches' and 'The prototype'; Jacopo Bono drafted 'State of the art: different approaches, between past and present' and 'Not only residential: collaboration with the Basic-Fit company'.

References

Alexander C. (1965). A city is not a Tree, Parts 1 & 2. In *Architectural Forum*, No. 122, pp. 58-62.

Barozzi Jacopo da Vignola (1562). *La Regola delli cinque ordini dell'architettura di Iacopo Barozzi parda Vignola*. Rome.

Bucci F., Mulazzani M. (2000). *Luigi Moretti: Opere e scritti*. Milan: Electa.

Friedman Y. (1971). Flatwriter: choice by computer. In *Progressive Architecture*, No. 3, pp. 89-101.

Goodfellow I., Pouget-Abadie, J., Mirza M., Xu B., Warde-Farley D., Ozair S., Bengio Y. (2020). Generative adversarial networks. In *Communications of the ACM*, Vol. 63, No. 11, pp. 139-144.

Heumann A., Davis D. (2020). Humanizing Architectural Automation: A Case Study in Office Layouts. In C. O. Gengnagel, J. Baverel, M. Burry, T. Ramsgaard, S. Weinzierl (Eds.). *7th Design Modelling Symposium "Impact: Design with All Senses"*. Conference proceedings. Berlin, September 2019, pp. 662-670. Cham: Springer.

Hu R., Huang Z., Tang Y., van Kaick O., Zhang H., Huang H. (2020). Graph2plan: Learning floorplan generation from layout graphs. In *ACM Transactions on Graphics (TOG)*, Vol. 39, No. 4, pp. 118:1-118:14.

Koning H., Eizenberg J. (1981). The language of the prairie: Frank Lloyd Wright's prairie houses. In *Environment and Planning. B: Planning & Design*, Vol. 8, No. 3, pp. 295-323.

Nauata N., Chang K. H., Cheng C. Y., Mori G., Furukawa Y. (2020). House-gan: Relational generative adversarial networks for graph-constrained house layout generation. In *Computer Vision-ECCV 2020: 16th European Conference. Conference proceedings*. Glasgow, UK, 23-28 August 2020. Part I 16, pp. 162-177.

Oxman R. (2017). Thinking difference: Theories and models of parametric design thinking. In *Design studies*, No. 52, pp. 4-39.

Para W., Guerrero P., Kell,y T., Guibas L., Wonka, P. (2021). Generative Layout Modeling using Constraint Graphs. *IEEE/CVF International Conference on Computer Vision. Conference proceedings*. Montreal, Canada, 11-17 October 2021, pp. 6690-6700. Computer Vision Foundation.

Stiny G. (1980). Introduction to shape and shape grammars. In *Environment and Planning. B: Planning & Design*, Vol. 7, No. 3, pp. 343-351.

Stiny G., Mitchell W. J. (1978). The Palladian Grammar. In *Environment and Planning B: Planning and Design*, Vol. 5, No. 1, pp. 5-18.

Wang H., Wang J., Zhao M., Zhang W., Zhang F., Guo M. (2018). Graphgan: Graph representation learning with generative adversarial nets. In *Proceedings of the AAAI conference on artificial intelligence*, New Orleans, Louisiana, USA, 2-7 February 2018, Vol. 32, No. 1, pp. 2508-2515.

Weber R. E., Mueller C., Reinhart C. (2022). Automated floorplan generation in architectural design: A review of methods and applications. In *Automation in Construction*, Vol. 140, pp. 104-385.

Wu W., Fu X. M., Tang R., Wang Y., Qi Y. H., Liu L. (2019). Data-driven interior plan generation for residential buildings. In *ACM Transactions on Graphics (TOG)*, Vol. 38, No. 6, pp. 1-12.

You J., Ying R., Ren X., Hamilton W., Leskovec J. (2018). Graphrnn: Generating realistic graphs with deep auto-regressive models. In *35th International Conference on Machine Learning*. Conference proceedings. Stockholm, Sweden, 10-15 July 2018, Vol. 80, pp. 5708-5717.

Zhang H., Goodfellow I., Metaxas D., Odena A. (2019). Self-attention generative adversarial networks. In *36th International conference on machine learning. Conference proceedings*. Long Beach USA, 10-15 June 2019, pp. 7354-7363.

Authors
Massimiliano Lo Turco, Politecnico di Torino, massimiliano.loturco@polito.it
Andrea Tomalini, Politecnico di Torino, andrea.tomalini@polito.it
Jacopo Bono, Politecnico di Torino, jacopo.bono@polito.it

To cite this chapter: Lo Turco Massimiliano, Tomalini Andrea, Bono Jacopo (2023). Un approccio euristico alla progettazione. Transizioni da algoritmi generativi a modelli parametrici/A Heuristic Approach to Design. Transitions from Generative Algorithms to Parametric Models. In Cannella M., Garozzo A., Morena S. (Eds.). *Transizioni. Atti del 44° Convegno Internazionale dei Docenti delle Discipline della Rappresentazione/Transitions. Proceedings of the 44th International Conference of Representation Disciplines Teachers*. Milano: FrancoAngeli, pp. 2914-2930.

 Isbn 9788835155119